

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»
Кафедра мультимедійних систем факультету інформатики

Розробка веб-ресурсу на основі принципу «Mobile-first»

**Текстова частина до курсової роботи
за спеціальністю «Інженерія програмного забезпечення»**

Керівник курсової роботи

доцент Афонін А. О.

_____ (підпис)

“ ____ ” _____ 2025 р.

Виконала студентка

Мельник М. О.

“ ____ ” _____ 2025 р.

Міністерство освіти і науки України

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»

Кафедра мультимедійних систем факультету інформатики

ЗАТВЕРДЖУЮ

Зав. кафедри мультимедійних систем,
доцент, кандидат наук

_____ Жежерун О. П.
(підпис)

“ ____ ” _____ 2025 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

на курсову роботу

студентці Мельник Марині Олегівні 3-го курсу факультету інформатики

ТЕМА: Розробка веб-ресурсу на основі принципу Mobile-first

Вихідні дані:

Використання принципу Mobile-first для покращення користувацького досвіду на мобільних пристроях на прикладі веб-ресурсу.

Зміст ТЧ до курсової роботи:

Індивідуальне завдання

Зміст

Анотація

Вступ

1. Теоретичні основи Mobile-first

2. Особливості розробки веб-ресурсів за принципом Mobile-first

3. Практична реалізація веб-ресурсу за принципом Mobile-first

Висновок

Список літератури

Дата видачі “___” _____ 2025 р.

Керівник _____ Завдання отримано _____

КАЛЕНДАРНИЙ ПЛАН ВИКОНАННЯ РОБОТИ

Тема: Розробка веб-ресурсу на основі принципу Mobile-first

№ п/п	ПЕРЕЛІК РОБІТ	Термін виконання	Примітка
1.	Вибір теми	Вересень – початок жовтня 2024	
2.	Затвердження теми з керівником	Жовтень 2024	
3.	Пошук та огляд літератури за темою роботи	Листопад – грудень 2024	
4.	Написання прототипу сайту з реалізацією основних принципів підходу Mobile-first	Січень – лютий 2025	
5.	Написання теоретичної частини роботи	Березень – квітень 2025	
6.	Подання на кафедру задля перевірки щодо вимог добročесності	5 травня 2025	
7.	Захист курсової роботи	12-14 травня 2025	

Студент Мельник М. О.

Керівник Афонін А. О.

« ____ » _____ 2025 р.

ЗМІСТ

КАЛЕНДАРНИЙ ПЛАН ВИКОНАННЯ РОБОТИ.....	4
ЗМІСТ.....	5
СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧКИ.....	6
АНОТАЦІЯ.....	7
ВСТУП.....	8
РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ MOBILE-FIRST.....	10
1.2 Альтернативні підходи: AWD та RWD.....	12
1.3 Переваги та недоліки Mobile-first.....	13
РОЗДІЛ 2. ОСОБЛИВОСТІ РОЗРОБКИ ВЕБ-РЕСУРСІВ ЗА ПРИНЦИПОМ MOBILE-FIRST.....	17
2.1 Основні принципи Mobile-first підходу.....	17
2.2 Методи та технології розробки за принципом Mobile-first.....	25
РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ ВЕБ-РЕСУРСУ ЗА ПРИНЦИПОМ MOBILE-FIRST.....	29
3.1 Концепція дизайну та засоби розробки.....	29
3.2 Демонстрація проєкту.....	31
ВИСНОВКИ.....	41
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	42

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧКИ

MFD – Mobile-first design, стратегія проектування інтерфейсу та розробки веб-сайту, за якої насамперед увага приділяється створенню версії для мобільних пристроїв;

UX – User Experience, досвід користувача, який він отримує при користуванні продуктом;

SEO – Search Engine Optimization, процес оптимізації сайту для покращення його видимості в результатах пошукових систем;

WCAG – Web Content Accessibility Guidelines, документ, що описує набір рекомендацій для розробки доступних інтерфейсів.

GUI – Graphical User Interface, графічний інтерфейс користувача, що дозволяє взаємодіяти з комп'ютерними програмами через візуальні елементи, такі як вікна, кнопки, іконки та меню.

АНОТАЦІЯ

Робота присвячена дослідженню підходу Mobile-first у веб-розробці, розглянуто його принципи, переваги та недоліки порівняно з альтернативними стратегіями адаптивної та респонсивної верстки.

Метою роботи є розробка та апробація методики створення веб-ресурсу за принципом Mobile-first для забезпечення високої якості користувацького інтерфейсу на мобільних пристроях. Для досягнення мети було проаналізовано сучасні тенденції мобільної веб-розробки, досліджено особливості Mobile-first підходу та реалізовано прототип сайту закладу харчування з урахуванням принципів Mobile-first.

Результатом роботи є детальний аналіз принципів Mobile-first підходу та розроблений адаптивний веб-ресурс, орієнтований насамперед на мобільні пристрої, з подальшою адаптацією для планшетів і десктопів.

Ключові слова: Mobile-first, адаптивний дизайн, веб-розробка, інтерфейс користувача, мобільні пристрої.

ВСТУП

У сучасному світі із розвитком мобільних пристроїв з'явилась нагальна потреба адаптації різноманітних веб-ресурсів до екранів різних розмірів для зручного доступу до веб-сайтів та додатків. Залежність користувачів від смартфонів та планшетів, що постійно зростає, змінила потреби у бік взаємодії із мобільними пристроями та, як наслідок, попит на веб-ресурси, що оптимізовані для менших екранів та сенсорної взаємодії, ще ніколи не був таким високим.

Підхід до дизайну, який орієнтований на мобільні пристрої, має переваги, гарантуючи, що певний додаток чи веб-сайт буде зручним для користування та доступним для користувача на широкому діапазоні пристроїв. В свою чергу, використання такого підходу дозволяє розробникам створювати більш привабливі та ефективні цифрові продукти, які забезпечують високий рівень зручності в перегляді контенту та навігації для мобільних користувачів.

Актуальність дослідження зумовлена стрімким збільшенням кількості користувачів мобільними пристроями та полягає в необхідності розробки веб-ресурсів, які адаптуються до різних розмірів екрану та є оптимізованими до забезпечення найкращого користувацького досвіду.

Метою роботи є розробка та апробація ефективної методики створення веб-ресурсу з використанням принципу «Mobile-first», що забезпечує високу якість користувацького інтерфейсу та оптимальну функціональність на мобільних пристроях.

Для досягнення поставленої мети передбачено вирішення наступних **завдань**:

- проаналізувати сучасні тенденції мобільної веб-розробки;
- дослідити принципи та методологію «Mobile-first» дизайну;
- визначити ключові критерії оптимізації веб-ресурсів для мобільних пристроїв;

- розробити власний веб-ресурс з дотриманням принципів «Mobile-first».

Об'єктом дослідження є процес проектування та розробки веб-ресурсів з використанням принципу «Mobile-first».

Предметом дослідження виступають методи та технології створення адаптивних веб-інтерфейсів, орієнтованих на мобільні пристрої.

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ MOBILE-FIRST

1.1 Поняття та важливість Mobile-first дизайну.

Mobile First Design (MFD) – це стратегія проектування інтерфейсу та розробки веб-сайту, за якої в першу чергу увага приділяється створенню версії для мобільних пристроїв, яка потім адаптується для екранів більших розмірів. Тобто особливість цієї концепції полягає в тому, що на початку роботи над веб-сайтом дизайнерами та розробниками створюється версія продукту для смартфонів або планшетів, і тільки після цього відбувається перехід до дисплеїв більших розмірів, таких як ноутбуки та десктопи.

Mobile-first базується на підході «прогресивний розвиток» («Progressive Advancement»), що означає розробку продукту, першочергово розрахованого на пристрої з великою кількістю обмежень. Така версія має набір найбільш необхідних функцій і може поступово збагачуватися новими опціями залежно від технічних можливостей інших пристроїв. Таким чином, створивши базовий макет для мобільного телефону, ми отримуємо сайт, який гарантовано працюватиме на планшетах, ноутбуках та ПК [1].

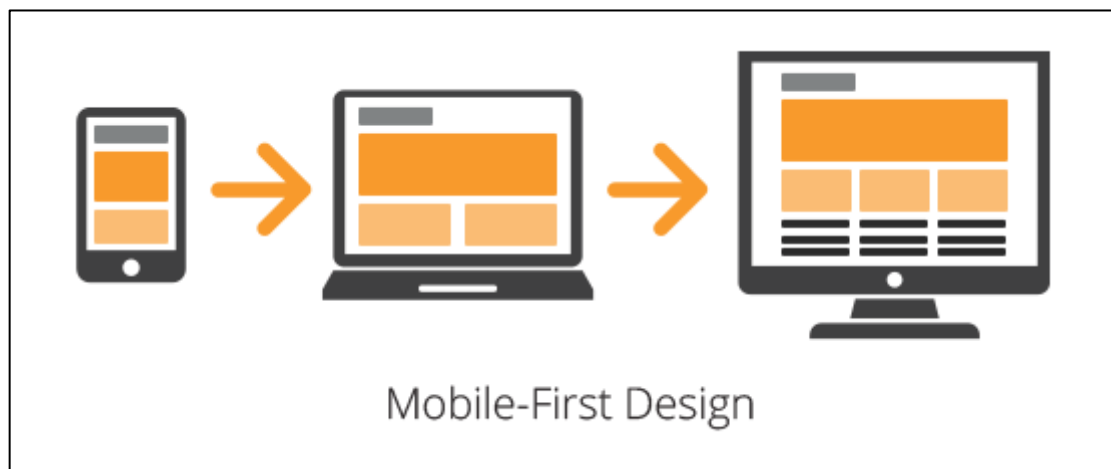


Рисунок 1.1 – Схема етапів адаптації інтерфейсу користувача згідно концепції Mobile-First

Впровадження дизайну, орієнтованого на мобільні пристрої, є дуже важливим через стрімке зростання використання мобільних пристроїв для відвідування інтернет-сайтів протягом останніх років. Відповідно до статистики відсоток відвідуваності веб-сайтів з мобільних пристроїв у всьому світі з 1-го кварталу 2015 року по 4-й квартал 2024 року зріс більше ніж на 30%. На мобільні пристрої припадає приблизно половина веб-трафіку в усьому світі. В останньому кварталі 2024 року мобільні пристрої (за винятком планшетів) згенерували 62,54% світового трафіку веб-сайтів [2].

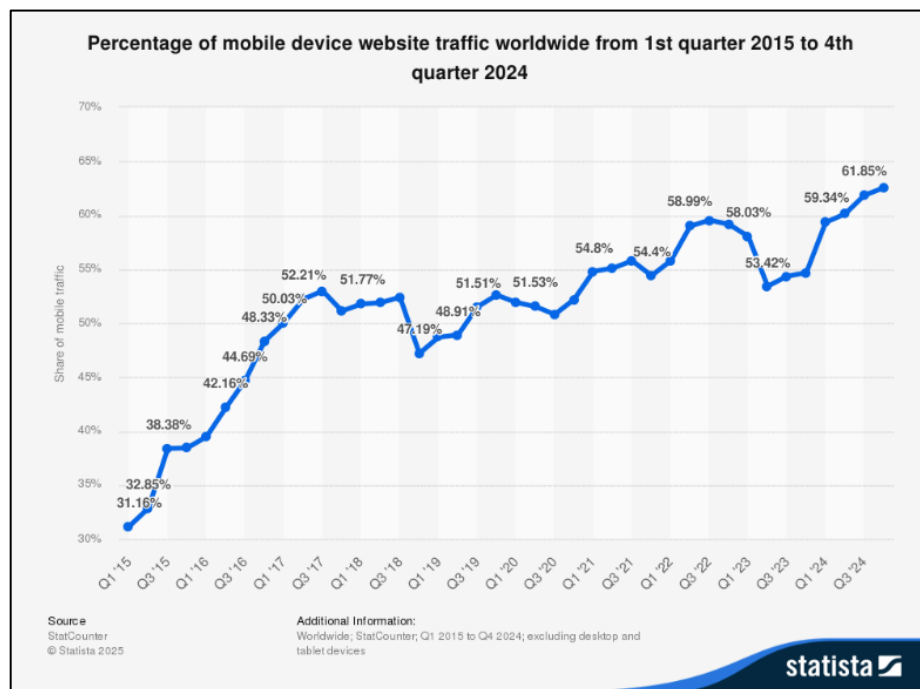


Рисунок 1.2 – Динаміка частки мобільного трафіку у світовому інтернеті з 1 кварталу 2015 року до 4 кварталу 2024 року

Враховуючи цю тенденцію, Mobile-first став стратегічно важливим підходом у веб-розробці. Він дозволяє створювати зручні, швидкі та ефективні веб-ресурси, адаптовані під сучасні потреби користувачів.

Проте стратегія Mobile-first не єдиний підхід до адаптації веб-інтерфейсів. У веб-дизайні існують також адаптивний (Adaptive) та респонсивний (Responsive) дизайн, кожен з яких має власні особливості, переваги та недоліки.

1.2 Альтернативні підходи: AWD та RWD

Adaptive web design (AWD) – це стратегія веб-дизайну, яка передбачає створення різних версій веб-сайту для різних пристроїв. В адаптивному дизайні, під час проектування веб-сайту, розробляється кілька статичних макетів для пристроїв різних типів. Зазвичай при проектуванні орієнтуються на розміщення елементів залежно від ширини пристрою. Створюють макети для кількох стандартних величин ширини, наприклад: 320px, 480px, 768px, 960px, 1200px, 1600px. За допомогою скриптів визначається розмір екрана користувача і відповідно до цього відбувається завантаження потрібного шаблону на підставі цих контрольних точок. Власне, адаптивний сайт змінюватиме розмір певним ривком, а також передбачає створення декількох окремих версій: звичайної (desktop), мобільної та версії для планшета, тобто веб-дизайнерам потрібно створювати три макета для одного сайту.

При використанні адаптивної стратегії дизайну зазвичай використовується підхід, який є протилежним до підходу Progressive Advancement – «витончена деградація» («Graceful Degradation»).

За стратегії Graceful Degradation розробка розпочинається зі створення версії для десктопів, після чого дизайн адаптується для екранів меншого розміру. Цей підхід часто веде до того, що мобільна версія є спрощеною копією десктопної. Така стратегія може негативно позначатися на функціональності та зручності використання сайтів на мобільних пристроях і призводити до погіршення користувацького досвіду.



Рисунок 1.3 – Схема поступової деградації інтерфейсу (Graceful Degradation)

Responsive web design (RWD) – це підхід, при якому веб-сайт автоматично адаптується до різних розмірів екрану, забезпечуючи безперебійну роботу на будь-якому пристрої. Респонсивний дизайн також називають чутливим, тому що він автоматично підлаштовується під пристрої, тобто можна змінювати масштаб сторінки і сайт має коректно відображатися. Головною перевагою Responsive Design є те, що не потрібно розробляти декілька версій сайту. Але при цьому потрібно ретельно перевіряти, чи коректно відображаються всі елементи на екранах різних розмірів.

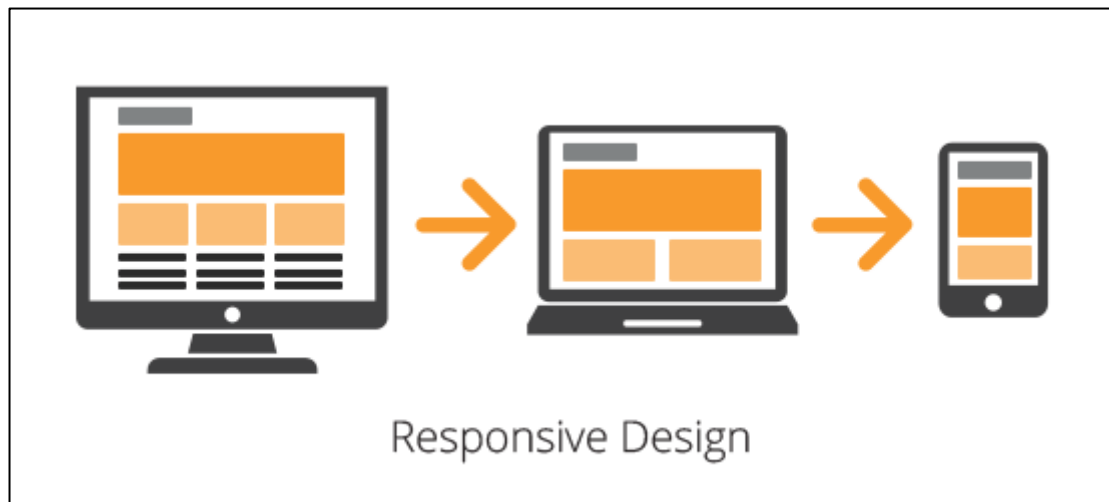


Рисунок 1.4 – Схема етапів адаптації інтерфейсу користувача згідно підходу Responsive design

І хоча обидва підходи до дизайну націлені на задоволення потреб користувачів і забезпечення безперешкодного доступу з будь-якого пристрою, стратегія Mobile-first вигідно вирізняється серед них завдяки ретельному плануванню та продуманому дизайну, що враховує специфіку мобільних пристроїв на кожному етапі розробки. Та попри це Mobile-first має власні переваги та недоліки.

1.3 Переваги та недоліки Mobile-first

Беззаперечно Mobile-first не є універсальним рішенням для розробки усіх веб-застосунків. У цьому розділі проаналізуємо усі його переваги та недоліки для

кращого розуміння складнощів, які можуть виникнути при його застосування у процесі розробки.

- Переваги принципу MFD:

1. Покращений користувацький досвід (UX)

Оскільки принцип Mobile-first першочергово орієнтується на мобільні пристрої, це гарантує що розробники та дизайнери зосереджуються на найважливішому контенті, роблять акцент на ключовій функціональності, забезпечуючи інтуїтивно зрозумілий інтерфейс та зручність використання сайту чи застосунку. Це в свою чергу значно покращує користувацький досвід, адже шлях користувача продумується та реалізується з урахуванням усіх обмежень, які є на малих екранах. Усі ці фактори підвищують задоволеність користувачів і збільшують ймовірність повторних відвідувань сайту.

2. Оптимізація для пошукових систем (SEO)

Починаючи з червня 2024 року Google оголосив про перехід на Mobile-First Indexing. Mobile-First Indexing – це процес, під час якого Google використовує мобільну версію веб-сайту як основну для індексації та ранжування. Отже, саме мобільний контент стає ключовим для визначення релевантності сайту у пошукових запитах. Варто зазначити, що, якщо мобільна версія сайту відсутня або вона не оптимізована належним чином, це може негативно вплинути на видимість ресурсу в пошукових системах [4]. Інші пошукові системи також надають перевагу в ранжуванні сайтам, які адаптовані під мобільні пристрої (зокрема Bing, DuckDuckGo та інші). Отже, якщо сайт створено за принципом Mobile-first це автоматично додає йому пріоритет у видимості на пошукових платформах.

3. Підвищена швидкість завантаження

Сайти, оптимізовані для мобільних девайсів завантажуються швидше та споживають менше даних. Це відбувається за рахунок мінімізації CSS та JavaScript коду, концентрації лише на найважливішому контенті та оптимізації завантаження (наприклад використання lazy loading). Швидкість завантаження є

важливим фактором для користувачів, особливо в умовах обмеженого інтернет-з'єднання.

4. Доступ до широкої аудиторії користувачів

Як було зазначено вище, кількість користувачів, які відвідують веб-сайти використовуючи мобільні пристрої постійно зростає, тому Mobile-first дизайн дозволяє охопити більшу аудиторію та залучити нових користувачів.

5. Ефективність розробки

Розробка спочатку для мобільних пристроїв дозволяє створити лаконічний та ефективний дизайн, у якому потім легше розширити функціональність та додати додаткові стилі для більших екранів.

- Недоліки принципу MFD:

1. Обмежений розмір робочого полотна

Оскільки дизайн повинен добре працювати на невеликих екранах, варіантів дизайну інтерфейсу може бути менше, якщо порівнювати з дизайном для десктопів. Також є потреба відмовитись від складних анімацій та інших креативних підходів у дизайні, адже їх використання сприятиме довшому завантаженню сторінки, перенавантаженню різними елементами та погіршенню користувацького досвіду.

2. Збільшення часу розробки

Оскільки веб-сайт повинен бути спроектований і розроблений таким чином, щоб добре працювати на різних розмірах екранів, створення сайту за принципом MFD може зайняти більше часу, ніж розробка однієї версії.

3. Ризик втрати функціональності

Зважаючи на обмеженість розмірів мобільних девайсів, деякі функції та можливості, у певних випадках, доведеться прибрати для таких пристроїв, щоб забезпечити хороший користувацький досвід.

4. Складніше тестування

Через наявність кількох пристроїв і різних розмірів екранів тестування веб-сайту розробленого за принципом MFD, може бути складнішим, ніж тестування однієї версії сайту.

Підсумовуючи даний розділ, можна зробити висновок, що Mobile-first підхід є хорошим інструментом у веб-розробці, який в умовах стрімкого зростання мобільного трафіку та вимог користувачів до швидкого та зручного доступу до інформації, стає не просто трендом, а необхідністю. Переваги MFD роблять його потужним інструментом для досягнення бізнес-цілей. Однак, необхідно враховувати і потенційні недоліки. Вибір між Mobile-first та іншими підходами до веб-розробки повинен базуватися на ретельному аналізі конкретних потреб проекту, його цільової аудиторії та доступних ресурсів.

РОЗДІЛ 2. ОСОБЛИВОСТІ РОЗРОБКИ ВЕБ-РЕСУРСІВ ЗА ПРИНЦИПОМ MOBILE-FIRST

Для правильної реалізації Mobile-first підходу, важливо дотримуватися певних принципів, які впливають на структуру, продуктивність і зручність використання сайту. Розглянемо які існують принципи MFD та рекомендації щодо їх реалізації.

2.1 Основні принципи Mobile-first підходу

1. Фокус на ключовому контенті

Оскільки мобільні екрани мають обмежений простір, важливо визначити найважливішу інформацію та зосередитися на ній. Люк Врублевській у своїй роботі [3] зазначає, що варто підводити людей до того, за чим вони прийшли, якомога швидше. Часто користувачі мобільних пристроїв не є повністю зосередженими на сайті, у таких ситуаціях ми маємо лише часткову увагу людей, тому їм потрібен чіткий, сфокусований дизайн, щоб виконати роботу, а не безліч варіантів навігації, що заважають. При проектуванні веб-сайту насамперед варто визначити необхідний мінімум елементів, які допоможуть користувачеві здійснити цільову дію, і тільки після цього переходити до додавання додаткового контенту.

2. Правильна візуальна ієрархія контенту

На мобільних пристроях контент має бути лінійним, з чіткою послідовністю подання. Тому найважливіша інформація має бути розміщена на центральній частині сторінки. Елементи другорядного значення можуть знаходитися як над, так і під основним контентом. А найменш пріоритетний, але все ж таки необхідний контент може знаходитись у верхній частині екрану.

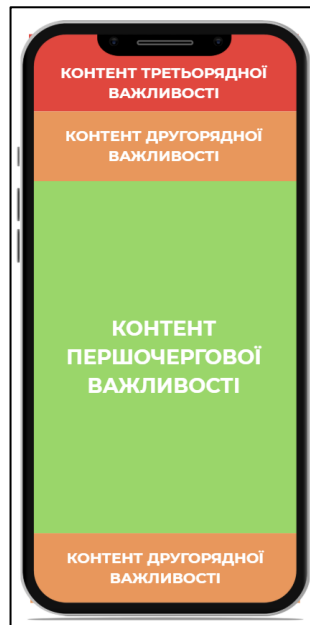


Рисунок 2.1 – Візуалізація пріоритетності контенту в мобільному інтерфейсі

Достатня кількість простору між елементами також покращує сприйняття інформації та допомагає виділити важливі частини. Тому важливо використовувати whitespace (негативний простір), щоб елементи не зливалися. Також для пріоритизації ключових компонентів необхідно використовувати короткі заголовки та чіткі СТА (Call-to-Action).

Одним із найважливіших принципів побудови візуальної ієрархії є розміщення основних інтерактивних елементів у найбільш доступних зонах екрана. Дослідження взаємодії з мобільними пристроями показують, що зона зручного натискання залежить від того, якою рукою користується людина. Так, для праворуких і ліворуких користувачів ці зони дещо відрізняються, проте загальна тенденція свідчить про те, що найбільш комфортною є нижня центральна частина екрана.



Рисунок 2.2 – Зони досяжності пальців на екрані смартфона при керуванні однією та двома руками

Саме в цій області рекомендується розташовувати основні елементи взаємодії, меню навігації та важливі інтерактивні блоки. Натомість верхня частина екрана є менш зручною для натискання, особливо під час використання пристрою однією рукою, тому важливі елементи не слід розміщувати там без нагальної потреби.

3. Простота навігації між сторінками

Інтуїтивно зрозуміла навігація дозволяє забезпечити чіткий і зрозумілий користувацький досвід на мобільних пристроях. За дослідженням [5] користувачам потрібно значно більше часу для виконання завдань на сайтах із прихованою навігацією, ніж на сайтах із видимою або комбінованою навігацією як на комп'ютері, так і на мобільному пристрої.

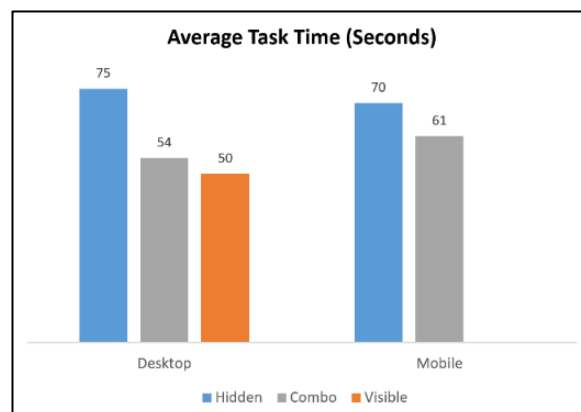


Рисунок 2.3 – Середній час виконання завдань на комп'ютерах та мобільних девайсах із різними типами навігації

Як зазначає Люк Врублевський у своїй роботі [3], мінімізація кількості варіантів навігації на екранах мобільних пристроїв допомагає запобігати помилкам. Адже чим менше варіантів навігації, тим з меншою ймовірністю користувачі випадково перейдуть до інших завдань, намагаючись досягти своєї поточної мети. Існують загальні рекомендації для навігації на мобільних пристроях: якщо сайт має 4 або менше навігаційних посилань верхнього рівня, варто показувати їх як видимі посилання, у випадку якщо ваш сайт має більше 4 навігаційних посилань верхнього рівня, рішенням є приховати деякі з них.

4. Адаптивність

Адаптивність у Mobile-first підході полягає в тому, щоб веб-дизайн і верстка забезпечували комфортну взаємодію з контентом на будь-якому мобільному пристрої, адже девайси бувають різних розмірів і важливо забезпечувати комфорт на кожному з них. Це означає, що текст повинен бути читабельним без необхідності масштабування, кнопки – достатньо великими для зручного натискання, а макет – не вимагати зайвих дій для навігації. Про правильність підбору типографіки буде описано в одному з наступних пунктів, що ж стосується інших клікабельних елементів, варто дотримуватись наступних рекомендацій: мінімальний розмір – 26 px, оптимальний – 40-44 px, а відступи навколо них – не менші ніж 10 px.

5. Мінімум дій користувача

Як вже зазначалось вище, для покращення користувацького досвіду варто впроваджувати функції, які пришвидшують виконання мети, з якою користувач відвідує сайт. При розробці веб-сторінок за принципом Mobile-first потрібно передбачати такі опції, призначені для полегшення шляху користувача [1]:

- Швидкий зв'язок через дзвінок/повідомлення/соцмережі – допомагає користувачу швидше зв'язатися з підтримкою або дізнатися додаткову інформацію.

- Автозаповнення полів – дозволяє швидше вводити контактні дані, адреси або платіжну інформацію. Це економить час і зменшує ризик помилок при введенні.
- Верифікація за біометричними даними – підтримка входу через Face ID, Touch ID або розпізнавання відбитка пальця дозволяє користувачам швидко й безпечно авторизуватися без необхідності вводити паролі.
- Геолокаційні можливості – автоматичне визначення місцезнаходження користувача для пропонування найближчих філій, розрахунку доставки чи підбору відповідного контенту.
- Автозбереження введених даних – якщо користувач випадково закриє сторінку або оновить її, введені дані (наприклад, у формі замовлення) не мають зникати.

Тут перерахована лише частина можливих покращень. Конкретний набір функцій залежить від типу сайту, його цільової аудиторії та завдань, які він виконує. Важливо аналізувати поведінку користувачів і впроваджувати ті рішення, які зроблять їх взаємодію із сайтом швидшою та зручнішою.

б. Інтуїтивна зрозумілість інтерфейсу

Користувачі повинні легко й без додаткових пояснень розуміти, як взаємодіяти з веб-сайтом. Розглянемо основні аспекти, яких варто дотримуватись, для забезпечення інтуїтивної зрозумілості:

- Прогнозованість – елементи інтерфейсу мають поводитися так, як очікує користувач. Наприклад, кнопка «кошик» у магазині повинна вести до перегляду покупок, «лупа» – до пошуку, а іконка соцмережі до відповідної соцмережі.
- Зрозумілість елементів – кнопки, іконки, форми мають бути візуально розпізнаваними й мати зрозумілі підписи. Використання загальноприйнятих символів та стилів допомагає уникнути плутанини.

- Зворотний зв'язок – сайт повинен реагувати на дії користувача (наприклад, змінювати колір кнопки при натисканні, відображати повідомлення про успішне відправлення форми).
- Мінімізація когнітивного навантаження – користувач не повинен витратити багато часу на розуміння структури сайту чи способів виконання базових дій. Простий, логічний і послідовний дизайн допомагає швидко зорієнтуватися.

Інтуїтивний інтерфейс підвищує ефективність взаємодії з сайтом, зменшує кількість помилок користувачів, а також покращує загальне враження від ресурсу.

7. Правильна типографіка

Текст веб-сторінок має бути легко читабельним як на найменших, так і на найбільших екранах. При розробці веб-сайту варто обмежити кількість використаних сімейств шрифтів до мінімальної кількості: двох, хоча в багатьох випадках буде достатньо і одного, та дотримуватися тих самих шрифтів на всьому проекті.

Розмір тексту має чи не найбільший вплив читабельність на сприйняття інформації і контенту на сайті. Для мобільних пристроїв мінімальний розмір шрифту для основного тексту повинен бути 16px. Заголовки рекомендується робити приблизно в 1,3 рази більшими за основний текст [6].

Рекомендованою довжиною рядка на мобільних пристроях є 30-40 символів, що робить текст легшим для сприйняття та читання. Міжрядковий інтервал має становити 120% від розміру шрифту. Також, для кращого сприйняття, текст варто вирівнювати за лівим краєм сторінки.

8. Уникнення складних анімації та спливаючих вікон

Як вже неодноразово зазначалось, мобільні пристрої мають обмежений простір, який звужує поле зору користувача при взаємодії з веб-сторінкою. Непотрібні спливаючі вікна та реклама, що займають весь екран, заважатимуть користувачеві працювати на сайті чи переглядати необхідну для нього

інформацію, що справлятиме негативне враження на користувача та сприятиме погіршення користувацького досвіду.

Що стосується використання складних анімацій, це теж зазвичай є невдалим рішенням для мобільних пристроїв. Використання складних CSS-анімацій або JavaScript-анімацій уповільнюють рендеринг сторінки, особливо на слабких пристроях.

Також це може викликати проблеми з доступністю веб-сторінки. Надмірне використання анімацій та спливаючих вікон може заважати користувачам із вадами зору або викликати дискомфорт у чутливих до руху людей.

9. Доступність

Доступність (Accessibility) у Mobile-first розробці означає створення веб-ресурсів, якими можуть комфортно користуватися всі люди, в тому числі й користувачі із фізичними чи когнітивними обмеженнями. Оскільки мобільні пристрої є основним засобом доступу до Інтернету для більшості користувачів, важливо забезпечити коректну роботу сайту для людей із вадами зору, слуху, моторики або іншими особливими потребами.

Ключові аспекти доступності в Mobile-first дизайні [7]:

- Кольоровий контраст має відповідати вимогам рівня WCAG 2.1 AA [8]:
 - Коефіцієнт контрастності 4,5:1 для звичайного тексту (менше 18 пунктів або 14 пунктів напівжирним).
 - Коефіцієнт контрастності 3:1 для великого тексту (принаймні 18 пунктів або 14 пунктів жирним).
 - Інформація, що передається за допомогою кольору, також має бути доступна іншими засобами (підкреслений текст для посилань тощо).
- Текстові еквіваленти:

- Описові текстові підписи (alt-атрибути) для зображень, щоб користувачі з порушеннями зору могли їх сприймати через екранні читачі.
- Транскрипція або субтитри для відео- та аудіоконтенту, що допоможе користувачам із вадами слуху.
- Загальні рекомендації:
 - Потрібно вказати назву програми/сайту.
 - Заголовки не повинні порушувати ієрархічну структуру.
 - Вміст не повинен обмежуватися однією орієнтацією, як-от портретною чи альбомною.
 - Об'єкти дотику мають бути достатньо великими, щоб користувач з ними міг взаємодіяти.

Розробка за принципом Mobile-first передбачає оптимізацію структури та функціональних елементів відповідно до обмежень мобільних екранів, що сприяє підвищенню швидкості завантаження сторінок, зменшенню зайвого контенту та спрощенню взаємодії. Особливу увагу слід приділяти зручності натискання, розташуванню основних елементів у досяжних зонах та забезпеченню доступу до основної інформації без зайвих дій. Таким чином, дотримання принципів Mobile-first не лише покращує юзабіліті сайту, а й підвищує його конкурентоспроможність, забезпечуючи комфортну та ефективну взаємодію користувачів із контентом.

Розглянувши основні принципи Mobile-first, варто перейти до практичних аспектів його реалізації. Розглянемо та проаналізуємо підходи до розробки, основні технології, що використовуються для створення адаптивних веб-ресурсів, а також приклади коду, які допомагають ефективно застосовувати Mobile-first у розробці.

2.2 Методи та технології розробки за принципом Mobile-first

Для реалізації підходу MFD використовуються різні методи та технології, які забезпечують адаптивність, продуктивність та зручність користування.

Основними з них є:

1. Використання адаптивної верстки

○ CSS медіа-запити (`@media`) – дозволяють застосовувати специфічні стилі на основі характеристик пристрою, таких як ширина, висота та орієнтація екрану. У контексті дизайну Mobile-first медіа-запити використовуються для визначення базових стилів для альбомних мобільних пристроїв, а потім поступово додаються стилі для більших екранів, забезпечуючи адаптивність.

```
/* Стили для мобільних пристроїв (за замовчуванням) */
body {
  font-size: 16px;
}

/* Стили для планшетів (ширина екрана від 768px) */
@media (min-width: 768px) {
  body {
    font-size: 18px;
  }
}

/* Стили для десктопів (ширина екрана від 1024px) */
@media (min-width: 1024px) {
  body {
    font-size: 20px;
  }
}
```

Рисунок 2.4 – Приклад коду з використанням медіа-запитів

○ Гнучкі макети: CSS надає зручні інструменти для створення гнучких і плавних макетів, які адаптуються до різних розмірів екрану. Використовуючи такі прийоми, як гнучкі сітки (Flexbox, CSS Grid), відносні одиниці (em, rem, vh, vw, %) та гнучкі зображення, ви можете створювати макети, які плавно підлаштовуються під розміри різних пристроїв, зберігаючи незмінний користувацький інтерфейс.

```
.elem-svg, .right-arrow, .search-svg, .left-svg {  
  width: 1.3rem;  
  height: 1.3rem;  
}
```

Рисунок 2.5 – Приклад коду з використанням відносних одиниць

2. Технології для забезпечення швидкості завантаження

Оскільки мобільні пристрої часто використовують нестабільні мережі, важливо мінімізувати час завантаження сторінки.

CSS дозволяє оптимізувати продуктивність за допомогою таких методів, як мінімізація CSS, вибіркоче завантаження стилів на основі можливостей пристрою та ефективного використання функцій CSS, таких як переходи та анімація. Використовуючи CSS для підвищення продуктивності, можна створювати легкі та швидко завантажувані таблиці стилів, які відповідають цілям продуктивності дизайну Mobile-first. Для завантаження зображень та контенту тільки тоді, коли вони потрібні варто використовувати ліниве завантаження (Lazy Loading).

```
<img className="rest-image"  
  src={restaurantImage}  
  alt="Restaurant Photo"  
  loading="lazy"  
>
```

Рисунок 2.6 – Приклад коду з використанням лінивого завантаження зображення

3. Фреймворки CSS

Основна ідея CSS-фреймворків (наприклад Bootstrap, Foundation) полягає в тому, щоб забезпечити стандартизований та узгоджений підхід до стилізації веб-сторінок. Вони містять попередньо визначені класи для типових елементів, таких як кнопки, форми, навігаційні меню та сітки, що дозволяє швидко створювати макети та застосовувати стилі без необхідності починати з нуля.

Однією з ключових переваг CSS-фреймворків є їхня здатність забезпечувати адаптивність. Вони часто включають гнучкі системи сіток та

медіа-запити, що дозволяє веб-сайтам автоматично адаптуватися до різних розмірів екранів та пристроїв. Це особливо важливо для розробки за принципом Mobile-first, де пріоритет віддається мобільним пристроям.

Однак важливо зазначити, що використання CSS-фреймворків також має свої недоліки. Вони можуть призвести до збільшення розміру файлів CSS, що може вплинути на продуктивність веб-сайту. Крім того, деякі фреймворки можуть бути надто великими або складними для невеликих проєктів.

4. JavaScript

JavaScript відіграє важливу роль у Mobile-first розробці, оскільки дозволяє забезпечити інтерактивність та ефективне управління контентом. Для зменшення затримок та покращення продуктивності важливо мінімізувати використання важких бібліотек, замінюючи їх на нативний JavaScript або легші фреймворки. Динамічне завантаження скриптів за допомогою `async` і `defer` дозволяє уникнути блокування основного потоку виконання. Крім того, правильна обробка подій, особливо для сенсорних екранів, дозволяє покращити взаємодію користувачів із веб-ресурсом.

JavaScript дозволяє використовувати прогресивні веб-додатки (PWA), які поєднують в собі переваги веб-сайтів та мобільних додатків. PWA можуть працювати в режимі офлайн, надсилати push-повідомлення та використовувати інші функції, доступні для мобільних додатків.

Таким чином, JavaScript є невід'ємною частиною розробки за принципом Mobile-first, дозволяючи створювати швидкі, адаптивні та зручні веб-інтерфейси, які забезпечують оптимальний досвід користувача на мобільних пристроях.

5. Тестування та налагодження Mobile-first сайтів

Однією з найважливіших складових Mobile-first розробки є тестування та налагодження, оскільки сайт має коректно працювати на широкому спектрі пристроїв.

Емулювання вбудованих у браузері інструментів, таких як Chrome DevTools чи Firefox Responsive Design Mode, дозволяє швидко перевірити

адаптивність макета. Однак тестування на реальних мобільних пристроях є незамінним, адже воно допомагає виявити можливі проблеми продуктивності та взаємодії, які можуть не проявлятися в емуляторах. Автоматизовані інструменти на зразок Lighthouse або PageSpeed Insights дозволяють оцінити швидкість завантаження сторінки, рівень доступності та відповідність SEO-вимогам, а кросбраузерне тестування за допомогою BrowserStack чи Sauce Labs допомагає перевірити сумісність веб-додатка з різними браузерами та операційними системами.

Підсумовуючи розділ, можна сказати, що основні принципи Mobile-first підходу передбачають впровадження правильної ієрархії контенту, реалізацію простої навігації між сторінками, фокус на ключовому контенті, дотримання принципів адаптивності та доступності, реалізацію інтуїтивно зрозумілого інтерфейсу та мінімізацію дій користувача. Кожен з цих пунктів має ключові аргументи для їх впровадження, а також веде до покращення зовнішнього вигляду та вдосконалення функціональності веб-ресурсу.

Використання адаптивного дизайну, гнучкої верстки, медіазапитів і сучасних технологій, таких як CSS Grid, Flexbox та прогресивне завантаження контенту, сприяє покращенню продуктивності та зручності взаємодії. Реалізація цих технологій дозволяє розширювати функціональність сайту для більших екранів без втрати якості користувацького досвіду.

РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ ВЕБ-РЕСУРСУ ЗА ПРИНЦИПОМ MOBILE-FIRST

У попередніх розділах було розглянуто основні принципи Mobile-first підходу при розробці веб-ресурсів. Цей розділ демонструє практичну реалізацію зазначених принципів шляхом створення прототипу власного сайту, орієнтованого насамперед на мобільні пристрої.

Оскільки ключовим аспектом Mobile-first підходу є зручність користувацького досвіду на невеликих екранах, особлива увага в розробці приділяється саме дизайну. Візуальна складова відіграє вирішальну роль у взаємодії користувача з сайтом, тому акцент цього розділу зроблено на адаптивності інтерфейсу, навігації та елементах взаємодії.

Створений веб-сайт є графічним інтерфейсом (GUI), що забезпечує можливість перегляду інформації, навігації, введення даних та отримання візуальної відповіді від системи.

3.1 Концепція дизайну та засоби розробки

За концепцію сайту було взято сторінку вигаданого закладу харчування «REST». Сайт для закладу харчування є оптимальним вибором для демонстрації принципу Mobile-first, оскільки більшість користувачів переглядають меню, контакти та здійснюють замовлення саме з мобільних пристроїв. Крім того, зручна мобільна версія покращує користувацький досвід, збільшує конверсію та підвищує конкурентоспроможність закладу в онлайн-середовищі.

Основні функціональні можливості, реалізовані в прототипі:

1. Перегляд меню – відображення категорій страв, їхнього опису, маси та ціни.
2. Замовлення страв – можливість обрати позиції з меню та сформулювати замовлення.

3. Контактна інформація – розташування закладу, номер телефону, години роботи, кнопка виклику.
4. Форма відгуків – поля для введення тексту та оцінки, інтерактивна кнопка надсилання.
5. Адаптивний дизайн – зміна розміщення елементів відповідно до ширини екрану.

Першим інструментом на шляху розробки став редактор інтерфейсів – Figma, що дозволило визначити основну структуру сайту, розташування елементів та взаємодію користувача з інтерфейсом. Прототип орієнтувався на мобільні пристрої, що відповідало принципу Mobile-first, а згодом був адаптований для більших екранів. Нижче зображено загальний вигляд макету в редакторі.

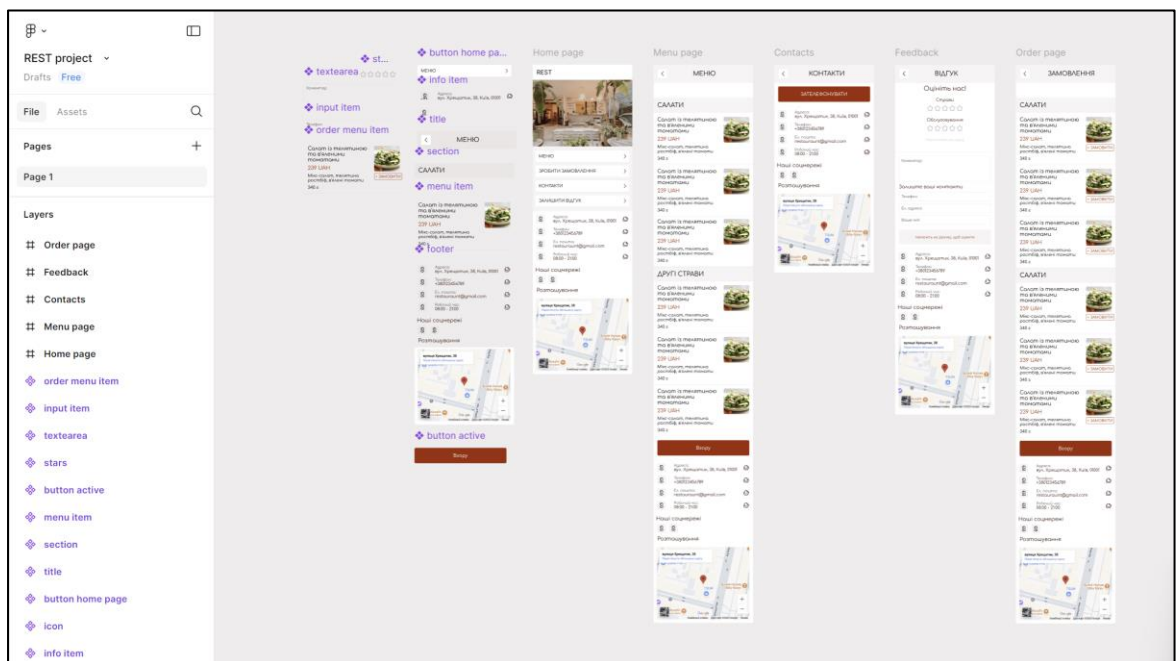


Рисунок 3.1 – Загальний вигляд макету сайту в редакторі Figma

Інструментами розробки даного прототипу є фреймворк JavaScript – React JS, а також застосування стилів CSS. Код зберігався у репозиторії на GitHub, що забезпечило контроль версій і зручну роботу.

3.2 Демонстрація проєкту

Розглянемо як саме було реалізовано принципи Mobile-first на прототипі веб-сайту для закладу харчування.

Кожна зі сторінок розробленого сайту має структуру колонки, що реалізує правильну візуальну ієрархію контенту. Ключовий зміст головної сторінки, сторінки відгуків та контактів знаходиться в центральній частині екрану з легким доступом для натискання як для праворуких, так і для ліворуких людей.

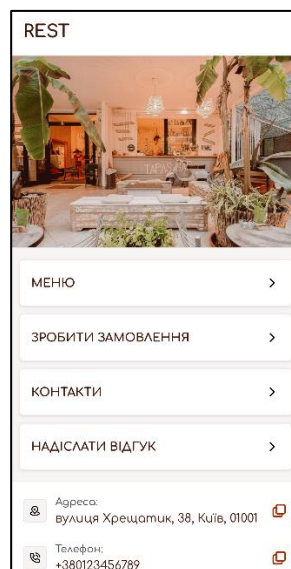


Рисунок 3.2 – Загальний вигляд головної сторінки

Кожна зі сторінок має однакову структуру:

- заголовок,
- ключовий зміст,
- футер.

Кожен із заголовків, сповіщає користувачеві про те, на якій із сторінок він знаходиться. Особливо важливим це є при створенні замовлення, оскільки цей процес складається з кількох етапів (вибір страв, введення контактних даних, підтвердження), важливо, щоб користувач завжди розумів, на якому кроці він перебуває.

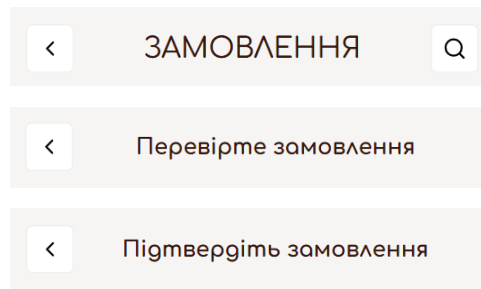


Рисунок 3.3 – Приклади заголовків різних сторінок сайту

Така реалізація спрощує навігацію, адже користувач швидко орієнтується в інтерфейсі, незалежно від того, на якій сторінці перебуває. Для Mobile-first підходу така уніфікованість особливо важлива, оскільки зменшує когнітивне навантаження на користувача, підвищує швидкість взаємодії з контентом, покращує адаптивність, оскільки однакова структура спрощує масштабування дизайну для різних пристроїв. Крім того, це позитивно впливає на UX, роблячи дизайн передбачуваним і комфортним для мобільних користувачів.

Наявність кнопки «Назад» (реалізована у вигляді стрілки у лівому верхньому куті) на всіх сторінках, окрім головної, забезпечує швидку та інтуїтивну навігацію, дозволяючи користувачеві легко повертатися до попереднього розділу. Це особливо важливо для мобільних пристроїв, де зручність взаємодії має ключове значення. Завдяки цьому сайт стає зрозумілішим і комфортнішим у використанні.

Єдиний футер на всіх сторінках забезпечує стабільність та зручність користувача. Він містить ключову інформацію: контактні дані, розклад роботи та інтегровану карту, що дозволяє швидко знайти необхідні відомості незалежно від того, на якій сторінці перебуває користувач.

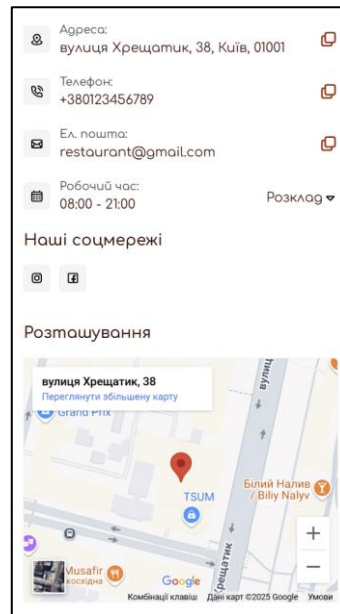


Рисунок 3.4 – Футер сайту

Як зазначалось у попередньому розділі ще одним ключовим принципом MFD є забезпечення мінімуму дій користувача. В прототипі це реалізовано наступним чином:

- швидкий зв'язок через дзвінок чи повідомлення – користувач має можливість з кожної зі сторінок зателефонувати до закладу або надіслати лист одним натисканням на контактний номер телефону або адресу електронної пошти. Нижче на фото наведено демонстрацію цих функцій.

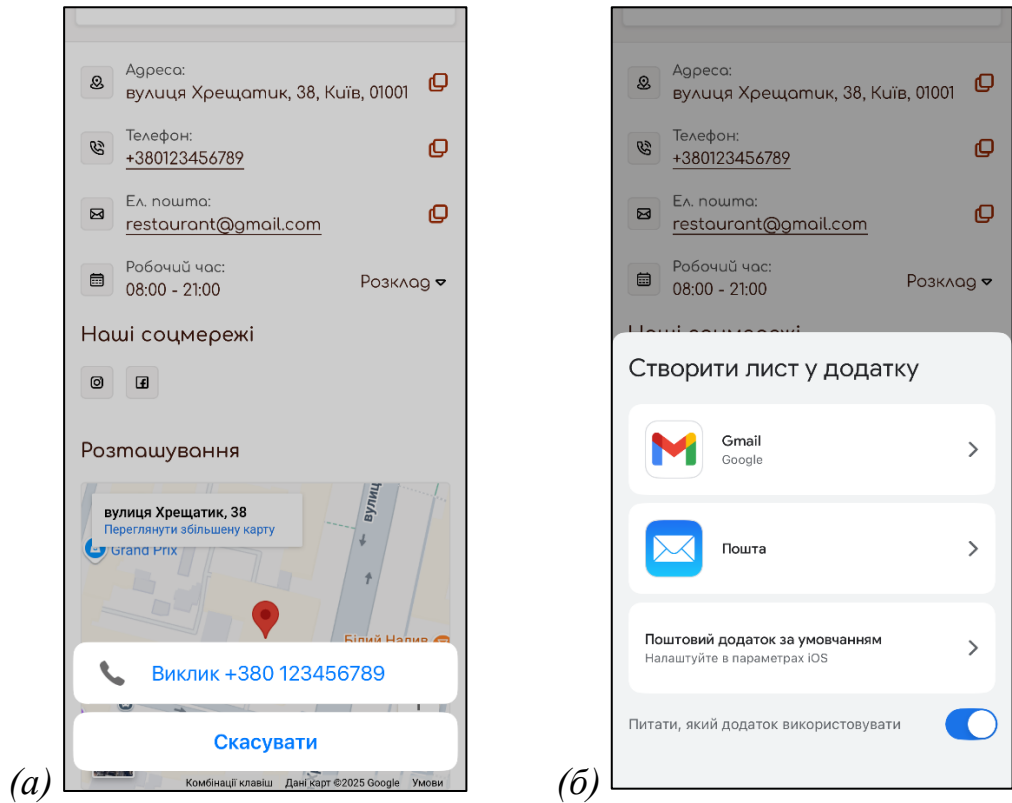


Рисунок 3.5 – Демонстрація функцій дзвінку до закладу (а) та надсилання листа (б)

Така ж функція реалізована на окремій сторінці контактів при натисканні на кнопку «Зателефонувати».

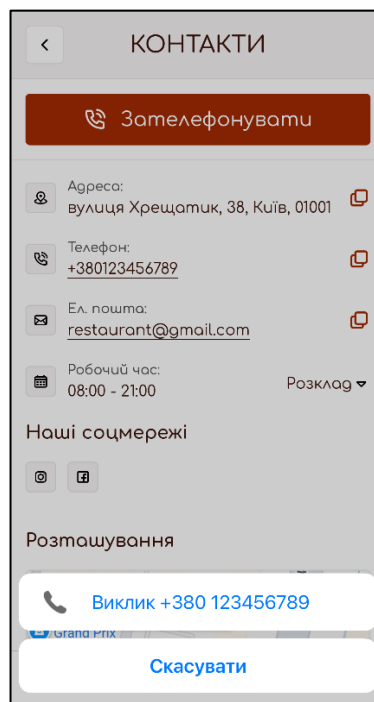


Рисунок 3.6 – Демонстрація функції дзвінку на сторінці контактів

○ швидкий доступ до геолокації закладу – користувач має можливість відразу на сайті переглянути розташування закладу, або ж одним натискання перейти у відповідний додаток (у випадку якщо потрібно прокласти маршрут тощо), де вже стоятиме мітка розташування закладу. Це допомагає уникнути зайвих дій, таких як копіювання інформації, перехід в інший додаток та пошук локації там.

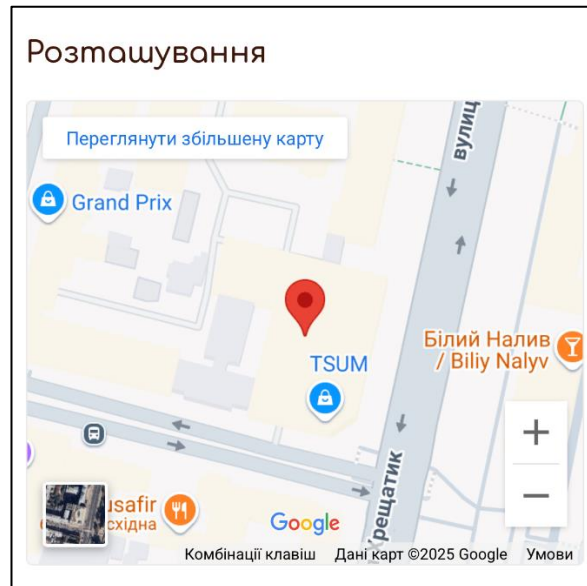


Рисунок 3.7 – Доступ до геолокації закладу

○ автозбереження даних – у випадку, якщо користувач під час введення даних або створення замовлення випадково оновить чи закриє сторінку усі дані зберезуться (дані зберігаються за допомогою localStorage). Це значно економить час користувача та убезпечує його від випадкових помилок.

```
orderItems [{"quantity":2,"item":{"id":1,"name":"Салат ...
  > {1: {quantity: 2, item: {id: 1, name: "Салат із телятиною та в'яленими тома
    > 1: {quantity: 2, item: {id: 1, name: "Салат із телятиною та в'яленими тома
    > 2: {quantity: 1, item: {id: 2, name: "Салат з лососем та авокадо", price:
```

Рисунок 3.8 – Приклад збереження даних замовлення у localStorage

Навігація сайту побудована таким чином, щоб бути максимально інтуїтивною для користувача мобільного пристрою. Переміщення між сторінками здійснюється покроково — лише вперед або назад, при цьому на

кожній сторінці є завжди видима кнопка повернення у вигляді стрілки та назва сторінки на якій знаходиться користувач, як вже зазначалось раніше.

Як зазначалось у попередньому розділі принципів MFD, якщо сторінка має 4 або менше навігаційних посилання, то їх варто показувати як видимі посилання, щоб користувачу не доводилось виконувати додаткові дії, такі як пошук прихованої навігації, її відкриття і лише потім перехід на потрібну йому сторінку. Саме таким чином реалізовані навігаційні посилання на головній сторінці, оскільки їх є 4, вони є видимими, знаходяться в центрі екрану, а отже є легко доступними для користувача.

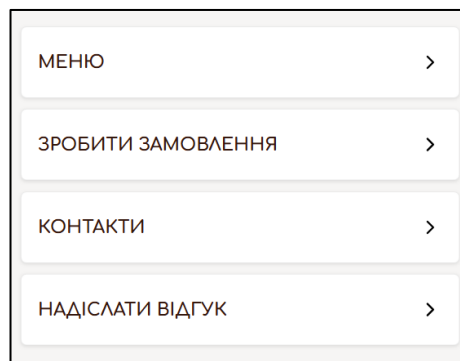


Рисунок 3.9 – Навігаційні посилання на головній сторінці

Що стосується навігації в інших частинах сайту, то на сторінках меню та замовлення реалізовано горизонтальний слайдер, який вказує на активний розділ і дозволяє швидко перемикатись між категоріями страв.



Рисунок 3.10 – Демонстрація розділів меню на сторінці меню

Також на цих сторінках реалізовано кнопку «Вгору», яка дозволяє швидко повертатись на початок сторінки, так як за рахунок великої кількості інформації на них, сторінки є досить об’ємними це корисна для користувача функція.



Рисунок 3.11 – Демонстрація кнопки «Вгору» на сторінці меню

Інтуїтивна зрозумілість сайту досягається в тому числі завдяки використанню знайомих іконок і передбачуваної поведінки елементів керування. Наприклад, кнопки зі знаками «+» і «-» дозволяють змінювати кількість товару, лупа виконує функцію пошуку, а смітник — видалення позиції із замовлення, а також інші елементи веб-інтерфейсу. Завдяки цьому користувачі легко орієнтуються на сторінці без необхідності в додаткових поясненнях.



Рисунок 3.12 – Демонстрація інтуїтивно зрозумілих іконок інтерфейсу

Одним із важливих аспектів реалізації mobile-first підходу стала типографіка, яка безпосередньо впливає на зручність читання з мобільних пристроїв. У прототипі було обрано один шрифт, що добре відображається на невеликих екранах – Comfortaa, із чіткою ієрархією заголовків та текстів для забезпечення візуальної структури контенту.

Типографіку було адаптовано до мобільних екранів згідно з рекомендаціями щодо зручності читання: базовий розмір шрифту встановлено на 16 пікселів, що є оптимальним для мобільних пристроїв. Для заголовків використано масштабування приблизно у 1.3 рази, що дозволяє зберегти чітку візуальну ієрархію без перевантаження інтерфейсу.

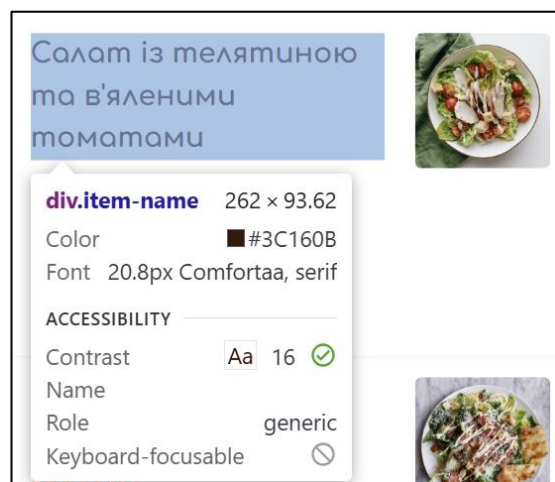


Рисунок 3.13 – Демонстрація типографіки заголовків

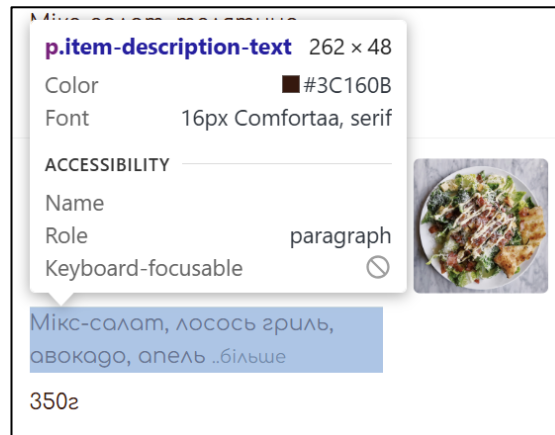


Рисунок 3.14 – Демонстрація типографіки основного тексту

Під час розробки також було враховано принципи доступності, щоб забезпечити комфортне використання сайту для ширшого кола користувачів. Зокрема, було підібрано достатній контраст між фоном і текстом, зрозумілі підписи до елементів інтерфейсу, додано описові текстові атрибути до зображень та реалізовано достатні розміри для клікабельних елементів.

Основними кольорами інтерфейсу було обрано темно-червоний та білий, які забезпечують високий рівень контрастності. Така комбінація має коефіцієнт контрастності 7.17 до 1, що відповідає підвищеному рівню доступності AAA, де мінімальне значення – 7:1, дозволяючи користувачам з порушеннями зору легко сприймати текстовий контент і навігаційні елементи.

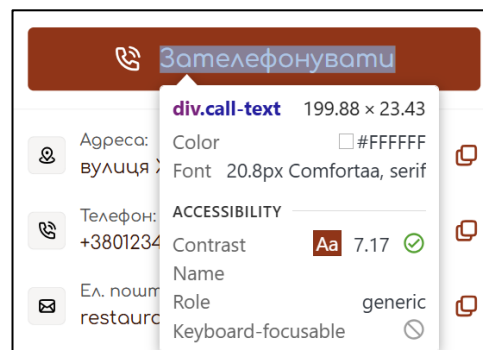


Рисунок 3.15 – Демонстрація рівня контрастності

Для всіх зображень на сайті було додано альтернативний текст (alt), що дозволяє користувачам із порушеннями зору або тим, хто використовує екранні рідери, отримати необхідну інформацію про вміст зображень.

```
<img className="rest-image" src={restaurantImage} alt="Restaurant Photo" />
```

Рисунок 3.16 – Приклад альтернативного тексту для зображення

Стосовно розмірів клікабельних елементів, вище зазначалось, що рекомендований розмір для них 40-44 пікселі, що також було реалізовано в прототипі.

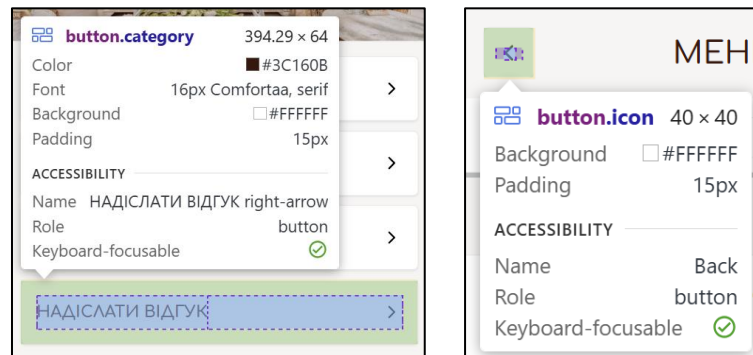


Рисунок 3.17 – Демонстрація розмірів клікабельних елементів

У підсумку, було продемонстровано концепцію сайту, засоби розробки, його структуру та здійснено повний огляд з демонстрацією усіх принципів Mobile-first підходу, яких було дотримано при розробці. Було показано реалізацію простої та зрозумілої навігації, забезпечення мінімальної кількості дій з боку користувача, підтримання єдиної структури на всіх сторінках, дотримання принципів доступності, правильної типографіки, а також створення інтуїтивно зрозумілого інтерфейсу. Таким чином, проєкт відповідає сучасним вимогам до зручного та доступного користувацького досвіду на різних типах пристроїв, та першочергово на мобільних девайсах.

ВИСНОВКИ

З кожним днем кількість користувачів мобільними пристроями зростає, а одночасно з цим зростає і відвідуваність веб-сайтів саме з мобільних девайсів. Саме тому забезпечення зручного веб-інтерфейсу на таких пристроях є актуальним та необхідним.

У роботі було досліджено теоретичні основи підходу Mobile-first, зокрема його сутність, переваги, недоліки, а також альтернативні стратегії адаптивної та респонсивної верстки.

Проаналізовано основні принципи підходу Mobile-first, такі як фокус на ключовому контенті, правильна візуальна ієрархія вмісту сайту, інтуїтивна зрозумілість інтерфейсу, правильна типографіка та інші. Обґрунтовано важливість їх застосування. Також розглянуто основні принципи та технології, що застосовуються під час розробки веб-ресурсів за цим підходом, серед яких медіа-запити, гнучкі сітки, адаптивна типографіка та оптимізація ресурсів для мобільних пристроїв.

Результатом роботи є реалізація основних принципів Mobile-first підходу на прикладі власного прототипу веб-сайту для закладу харчування. Спочатку було створено інтерфейс для невеликих екранів з урахуванням усіх описаних в роботі рекомендацій для забезпечення привабливого, а головне зручного інтерфейсу користувача, після чого здійснено адаптацію дизайну для планшетів та десктопів.

Отримані результати підтверджують ефективність Mobile-first підходу для створення сучасних, швидких і зручних веб-ресурсів, що відповідають потребам користувачів на різних пристроях.

У майбутньому це дослідження варто розширювати, додаючи реалізацію серверної логіки та збільшуючи функціонал сайту. Додатково, цікавим напрямом є вивчення впливу Mobile-first підходу на продуктивність, SEO та юзабіліті, що дозволить ще краще адаптувати веб-ресурси до потреб сучасних користувачів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Mobile First Design при створенні сайтів: що він означає і чому є настільки актуальним. E Sputnik.com: веб-сайт. URL: <https://esputnik.com/uk/blog/mobile-first-design-pry-stvorenni-saitiv-shcho-vin-oznachaie-i-chomu-ie-nastilky-aktualnym>
2. Percentage of mobile device website traffic worldwide from 1st quarter 2015 to 4th quarter 2024. Statista.com: веб-сайт. URL: <https://www.statista.com/statistics/277125/share-of-website-traffic-coming-from-mobile-devices/>
3. Wroblewski L. Mobile first. New York: A Book Apart, 2011. 132 с.
4. Mobile-First Indexing: що це таке та чого очікувати власникам сайтів? Wedex.com.ua: веб-сайт. URL: <https://wedex.com.ua/blog/mobile-first-indexing-shho-cze-take-ta-chogo-ochikuvaty-vlasnykam-sajtiv>
5. Hamburger Menus and Hidden Navigation Hurt UX Metrics. Nngroup.com: веб-сайт. URL: <https://www.nngroup.com/articles/hamburger-menus/>
6. Типографіка в мобільних додатках: поради та рекомендації. Komarov.design: веб-сайт. URL: <https://www.komarov.design/tipoghrafika-v-mobilnikh-dodatках-poradi-ta-riekomiendatsiyi/>
7. Mobile accessibility checklist. Developer.mozilla.org: веб-сайт. URL: https://developer.mozilla.org/en-US/docs/Web/Accessibility/Guides/Mobile_accessibility_checklist
8. Web Content Accessibility Guidelines (WCAG) 2.2. W3.org: веб-сайт. <https://www.w3.org/TR/WCAG/#contrast-minimum>